COVER SHEET

A DIGITAL ARTWORK EXPRESSION LANGUAGE (DAEL)

Robert Smith, Binh Pham and Sharmin Choudhury (Tinni)
Queensland University of Technology
Brisbane, Australia
r2.smith@qut.edu.au, b.pham@qut.edu.au, t.choudhury@qut.edu.au

ABSTRACT

To provide effective representation, search and retrieval of digital artworks, their semantics, context and symbolism should be considered. Existing metadata schemas were designed for other specific purposes and are too cumbersome to be modified to express these abstract concepts of digital artworks. We have designed a new metadata wrapper schema suitable for digital artworks and a Digital Artwork Expression Language (DAEL) based on this schema. This paper focuses on the requirements and features of DAEL, and how it can be used to express basic metadata as well as to generate more complex and abstract metadata via a Transformation Engine. A use case is presented to illustrate the usefulness of this language.

KEY WORDS

Digital Artworks, Metadata, Ontology, Semantics

1. Introduction

As artworks are increasingly stored in digital forms, there is an impetus to develop intelligent software systems and languages to improve their representation and management, and thus widen their access. This necessitates the development of techniques that facilitate the incorporation of human expertise, knowledge and requirements. Currently, digital artworks are predominantly stored in forms that are unsuitable for search and retrieval based on high level semantics or context. This causes difficulty when users wish to access these works for cultural or historical studies through concepts such as artists' intent, emotions, themes, symbolism, or historical contexts. A lot of content is therefore locked away and inaccessible because existing metadata schemes do not suitably describe the artwork. There is no prior work on artwork expression languages that aim to facilitate access to this abstract information.

While Dublin Core [1, 2] can adequately cater for simple and precise information required for cataloguing such as basic technical and administrative data, it does not provide ways to represent the more abstract elements of artworks. Later schemes such as METS [3] and MPEG-7 [4] provide richer structural frameworks for metadata to ensure that digital objects in library collections are preserved, but they still have shortcomings when handling abstract concepts and themes used to connect artworks – such as peace or war. Furthermore, METS was designed for archiving and so has mandatory fields that are irrelevant and that also make it cumbersome and inefficient to implement for other purposes [5].

We have designed an appropriate metadata schema and an expression language that can be used to describe digital artworks in various domains (museums, home photos, films etc.). Our aim is to provide more meaningful ways for classification and retrieval of digital artworks. We specifically want to support high-level semantic queries that may be abstract or symbolic in nature. Our intention is to make the metadata wrapper schema very light weighted by minimizing the number of mandatory fields and reducing the depth of hierarchical levels. Details on this metadata wrapper schema are presented in an accompanying paper [6]. This paper focuses on the Digital Artworks Expression Language (DAEL) which was developed based on our metadata wrapper schema. The aims of this language are three-fold: (i) to provide a convenient and comprehensive means to describe an artwork; (ii) to communicate with the system in order to generate high-level abstract data from low-level metadata and other information; and (iii) to express input and output of queries.

In Section 2, we describe the required features of the DAEL and present an example application to a digital artwork. Section 3 gives a brief overview of the metadata schema used by the DAEL and describes in particular a module extension to the schema. We then present in Section 4 the DAEL features and its ability to transform the metadata content into various forms – abstract and real. Section 5 outlines the DAEL implementation and Section 6 concludes with an overall evaluation and discussion on potential further applications.

2. Requirement Analysis

In this section, we will discuss the required features of the DAEL and metadata scheme, as well as the data types that the DAEL will support. An example of a digital artwork will be used to illustrate these concepts.

2.1 Features

There are three essential features that the DAEL and metadata schema must have for them to be useful for managing digital artworks in a flexible way.

Firstly, there must be interoperability with other metadata schemes (such as Dublin Core). It is important that our schema can ingest the data that may already be available for a particular digital artwork.

Secondly, it must support association rules and cross referencing to allow references to other digital artworks that are related – perhaps by a particular artist, style or theme.

Lastly, it must be modular and extensible. There must be parts of the schema that can be extended to increase the functionality of the DAEL. Likewise, there must be a mechanism for adding new modules to the DAEL in order to handle new data types or new user requirements. Some of the data types that can already be managed are described in the next section.

2.1 Data Types

There are a number of simple data types that the DAEL must be able to support:

- Content metadata – these are the objects that are represented in the digital artwork or the component parts that comprise the artwork.
- Technical metadata – these are the digital format details such as the media, file types, compression algorithms etc.
- Associative metadata – these are the links to other artworks and are used to cross reference with similar artworks by links such as artist, style, or content themes.
- Administrative metadata – this is the more basic information regarding ownership, location, availability, rights, etc.

Then there are the more complicated and abstract data types that also must be supported by the DAEL and schema:

- Artwork Semantics – this is a description of the meaning that can be found in the artwork contents and how they are arranged. We use a *semantic ontology* to make the transformations between content and semantics.
- Symbolism – these are the new meanings that can be found by applying a domain specific *symbol thesaurus* to the digital image content and semantics. It allows the user to discover other messages intended by the artist, perhaps imparted by references to established cultural icons at the time creation.
- Context transforms – these are the digital artwork meanings under different contexts. Context is about location and presentation of the artwork such as a museum installation, a documentary film, a holiday photo etc. The transforms are made using the content as the input as well as other metadata available about the digital artwork – such as production notes, distribution, utilisation and rights.

2.2 Use Case

To illustrate the concepts described in the last subsection, we consider a digital image of a traditional Vietnamese painting called “Honours” (as shown in Figure 1). We can deconstruct its content and potential interpretations in various ways.

Firstly, there is the image content, which broadly speaking comprises of a boy, a rooster and a lotus flower. The semantics of the content could be described as a small boy besides a lotus flower and holding a rooster. The interpretation will depend on the relative locations of the objects within the image and the sophistication of the content description.

Figure 1 – A Traditional Vietnamese artwork

The next transformation would be the symbolism that can be attributed to this content and semantics. If we take the symbolic icons of the time, we know that the rooster depicts strength or courage and a lotus flower represents longevity. The boy can be inferred in this context to represent fertility.

Thus, we would like to be able to search for this image using content (lotus flower), or semantics (boy holding rooster), or symbolic meaning (a wish for offspring, strength and longevity). These are the types of search scenarios that our metadata schema and DAEL can support.

3. Metadata Schema Design

For a related project, we have developed a metadata wrapper schema called Loculus [6] which was designed to wrap around a wide variety of information from the different stages of life of an artefact; as well as such information that aids in the creation, use and preservation of said artifact. Loculus was also designed to house the mechanisms by which the history of the artefact can be stored. In addition, Loculus may also store algorithms that can perform certain dynamic functions. Loculus is extensible and is lightweight in the sense that the number of mandatory fields are kept to a minimum. Therefore it is the perfect choice to wrap and support our digital artwork metadata and then interface with our DAEL.

3.1 The Metadata Wrapper Schema

The overall wrapper schema design is as shown in Figure 2. The artefact element contains the schema extensions for the content as well as the other transforms facilitated by the DAEL. There are several associated elements that store the production, distribution and other administrative aspects of the digital artwork.

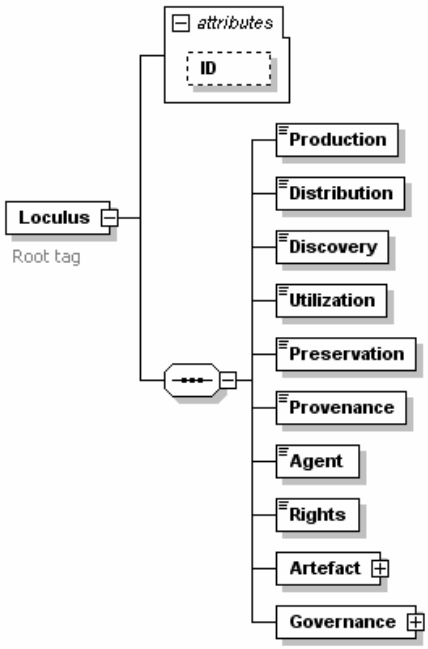

Figure 2 – The structure of the Loculus schema

The Loculus metadata wrapper has been made to be easily extended in a modular way. We have designed the extension for the Artefact element for use with our DAEL. The rest of the Loculus elements (Production, Distribution, Discover, etc.) can also be used by the DAEL abstraction transformations. The elements of the extension are shown in Figure 3 below. This paper focuses on this extension, rather than the whole metadata schema wrapper.

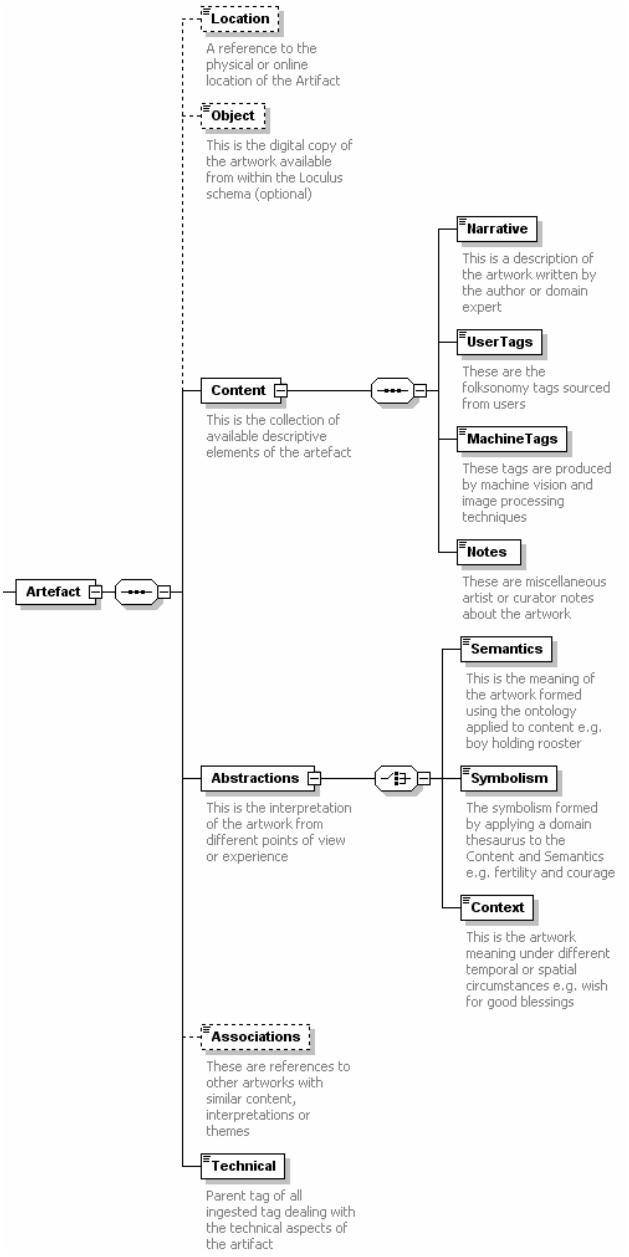

Figure 3 – The structure of the Artefact extension

Most of the extension elements in Figure 3 are self explanatory but we will elaborate on the basic fields within the Content element:

- Narrative – these are the abstracts or stories associated with the digital artwork. The narrative is usually written by an expert (or the artist) that associates a story along with the work. Sometimes it is a snapshot of a story that the artwork is about.
- User Tags – these are tags that are made available about the artwork through a folksonomy (such as FlickR [7]) or other such aggregation of user labels.

- Machine Tags – these are the computer identified content within the artwork – such as objects (boy, flower), colours, textures, and spatial relationships. We have demonstrated that such content can be automatically extracted from the artwork using computer vision techniques [8].
- Notes – these are the production notes by the artist, comments by experts (such as the curator or editor) and the formal labelling of the artwork according to various taxonomies and categorization systems.

These Content elements provide the input that is used by the Transformation Engine to generate the abstractions from the content. These are the Semantics, Symbolism and Context abstractions already described in Section 2.1. The next section describes the transformation mechanism and gives an example.

4. Digital Artworks Expression Language

The DAEL has been developed primarily to provide a comprehensive way for describing digital artworks. This language can also allow the interaction with an appropriate metadata schema module in order to produce more complex and abstract metadata for a digital artwork via the Transformation Engine. Once such high level metadata are produced, they can be used as indices to facilitate better search and retrieval. Given its simple and concise structure, the DAEL can also provide an efficient and effective mechanism to communicate the input and output of queries between users and the system.

4.1 Transformations

We have identified three transformations for the artwork content (and there is the provision to add more). These are the Semantic, Symbolic, and Contextual abstraction transformations. The relationships between the Content metadata and the other DAEL abstraction transformations are shown in Figure 4.

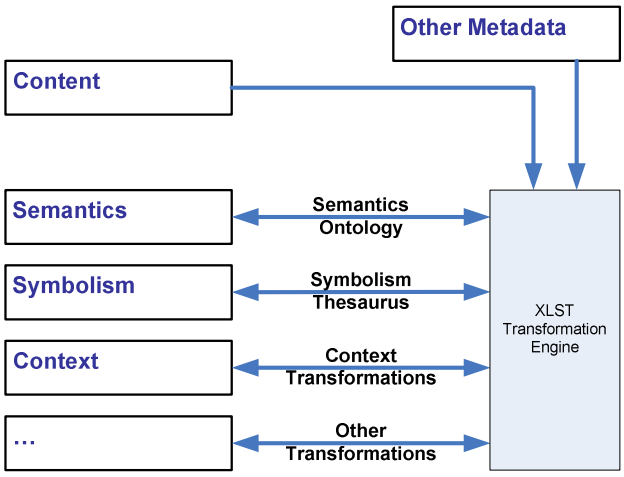

Figure 4 – Potential transformations in the DAEL.

These abstractions also have specified dependencies that must be adhered to in order to properly resolve any queries (see Figure 5). For example, symbolism will depend on not just the content of the artwork, but also the semantics (e.g. the way the content is structured and the relative meaning of the content positioning within the structure) *and* the Context (e.g. when it was created and for what purpose). The Context is the setting of the artwork and affects the Symbolism transformation accordingly.

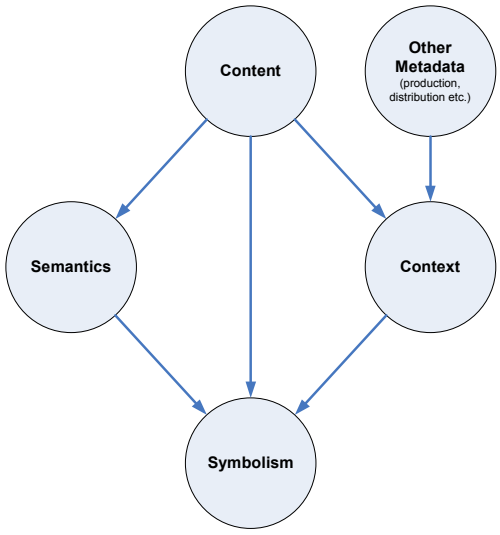

Figure 5 – The current transformation dependencies within the DAEL.

4.2 An Example

There are several descriptive elements and transformation functions that can be associated with the example digital artwork (Figure 1) described in Section 2.2. We illustrate the transformations and their outcomes for that use case in Figure 6 below.

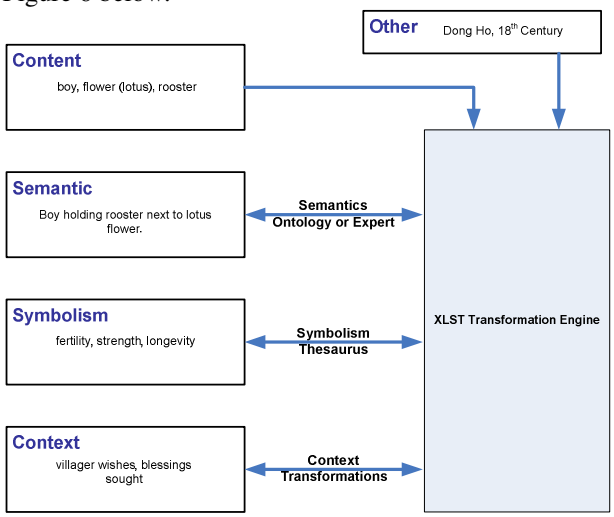

Figure 6 – An example of the transformations and their interaction for the use case.

The artwork content can be sourced in various ways – from the artist's notes, expert observations, curator notes, user tags or machine detected objects. From these sources, we can acquire the semantic transformation using the semantics ontology.

Then the context is determined, based on the semantics and other administrative metadata available on the artefact. This includes production notes, where and when it was made, the artist and the style of the artwork. This forms the Context abstraction.

Next, the Symbolism abstraction is formed by transforming the combination of the artwork content and semantics using a symbolism thesaurus that is applicable given a certain Context. For example, it may be important that the boy is holding the rooster – which will convey a different meaning to if the rooster was roaming free. So the symbolic abstractions can be formed from the knowledge that the painting was produced during the preparation for the Lunar New Year and therefore was intended to convey a wish for good blessings.

These transformations can then be used to classify and facilitate retrieval based on abstract queries. A possible scenario might be that the user is looking for paintings that portray good blessings. We can search for "good blessings" and return all metadata contains this search term in the transformation elements. In addition, we can return more information about the abstraction by using the DAEL transformation dependency structure. In this way the DAEL specifies the transformations and their dependencies to produce more meaningful return results.

The structure of the return result has the following format:

Artwork contains **Content** with **Semantics** symbolising **Symbolism** in **Context.**

For example, the return value for the use case would be: Painting "Honours" contains "boy, rooster, lotus flower" where "boy is holding rooster next to lotus flower" which symbolises "a wish for good blessings of fertility, courage and longevity" in "18th Century Vietnamese village life".

5. Implementation

The implementation to create the extensions can be written in any language that can handle and transform XML marked up data, but we have chosen to use XSLT [9] which facilitates easy referencing and accessing of the transformation instructions from within the Loculus document.

The transformation modules are available to the metadata schema by referencing the location of the XSLT style sheet. For every transformation, there is a reference to a different style sheet.

Each module acts on the current metadata schema document and appends additional data elements as the transformations are executed. If they do not already exist, the transformations are dynamically executed when a request to view that particular transform element is received (for search or browsing purposes).

5.1 XSLT Transformation Example

The Transformation Engine uses XSLT style sheets [] to effect these transformations. XSLT is the most widely used XML transformation language. It uses a marked up style sheet to specify the nature of the transformations. XSLT has all the traditional benefits of a high-level declarative programming language, specialized to the task of transforming XML documents.

Figure 7 illustrates the design of the MachineTags element within the Content element (see Figure 3) – an example of one part of the metadata extension to Loculus that is used by the XSLT transformations to form abstractions with.

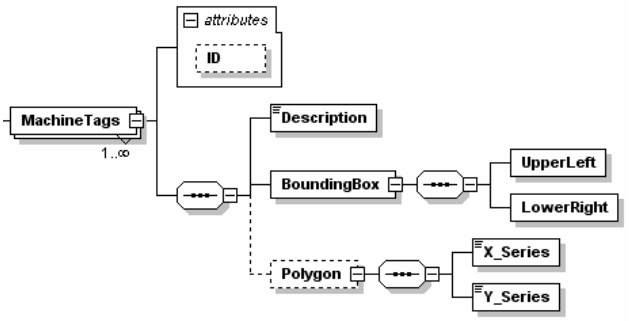

Figure 7 – The MachineTags element extension to Loculus.

The input elements are in XML format, and the transformation applies the appropriate XSLT style sheet to produce the output abstraction (also in XML). In Figure 8 we show the full mechanism for the creation of the Symbolism abstraction.

In this diagram, the Symbolism transformation uses three input elements – Content, Semantics and Context. The transformation mechanism is the XSLT style sheet and it uses a symbolism thesaurus (in a database form) to lookup the meaning of objects in the Content. It also examines the Semantics data for any extended meaning for objects according to their interaction with *other* objects. The Context metadata is then used to qualify the symbolic references to ensure that they are still meaningful under different domain contexts. Semantic and Context transformations are somewhat simpler.

After the transformation has been applied, the new Symbolism abstraction element is then available for classifying and retrieving the digital artwork.

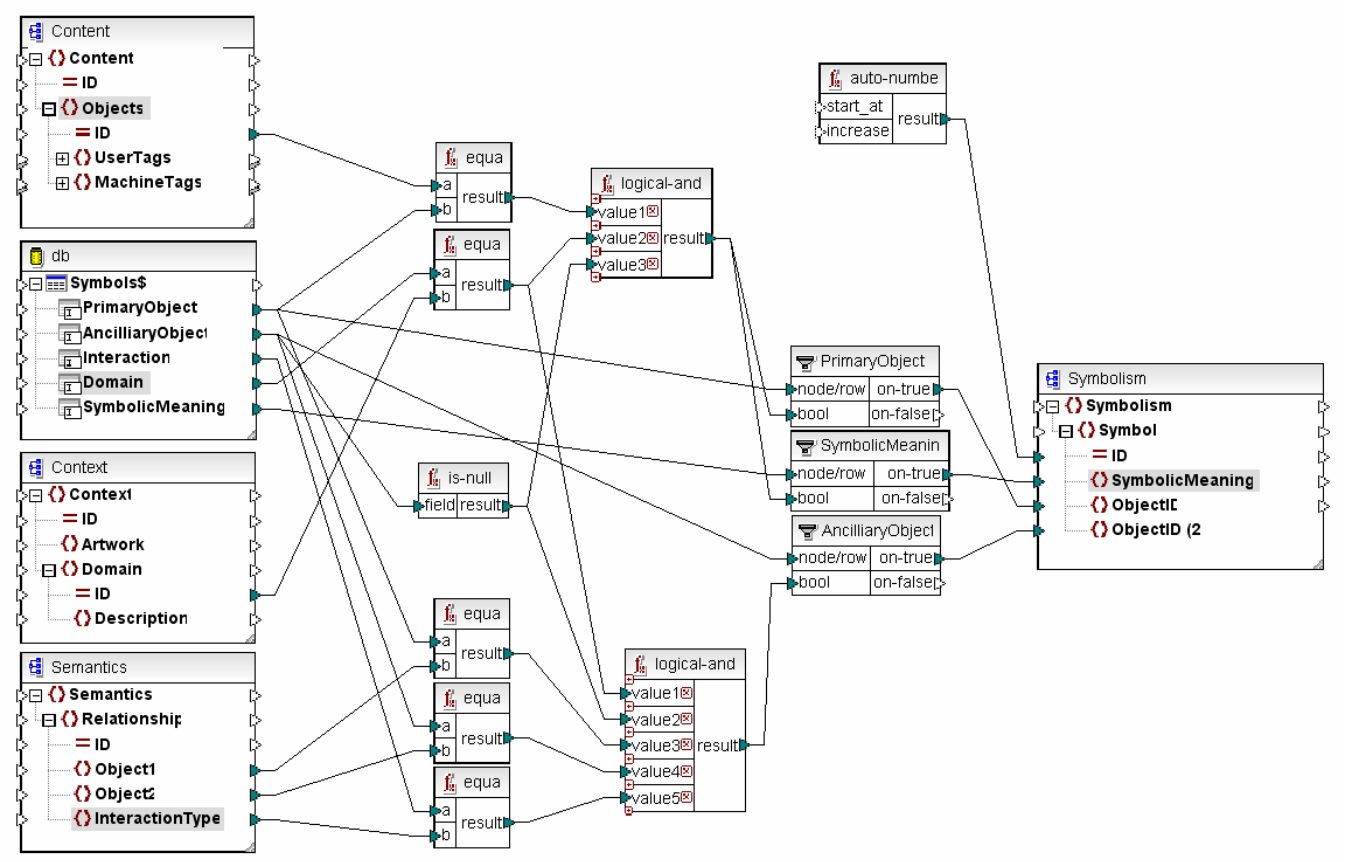

Figure 8 – The transformation for Symbolism.

6. Conclusion

We have developed a Digital Artworks Expression Language, designed to work in conjunction with a lightweight metadata wrapper schema Loculus. This language will enhance the management capability of digital artworks as well as broadening their access via more meaningful and abstract metadata in terms of semantics, context or symbolism. We will continue to evaluate both the Loculus and DAEL further through the construction of appropriate domain ontologies and thesaurus; and integrating them with different types of content to be extracted by computer vision techniques.

Acknowledgements

The authors thank the support of the ARC Centre of Excellence for Creative Industries and Innovation (CCI).

References

1. DCMI. Dublin Core Metadata Initiative. 2007 [cited 2007 March]; Available from: http://dublincore.org/documents/1999/07/02/dces/.
2. Riley, J. and A. Hutt, Semantics and syntax of dublin core usage in open archives initiative data providers of cultural heritage materials, in Digital Libraries, *2005. JCDL '05. Proceedings of the 5th ACM/IEEE-CS Joint Conference on*. 2005. p. 262--270.
3. METS: Metadata Encoding & Transmission Standard. 2007 [cited 2007 March]; Available from: http://www.loc.gov/standards/mets/.
4. MPEG. The MPEG Home Page. 2007 [cited 2007 March]; Available from: http://www.chiariglione.org/mpeg/.
5. Kim, W., On Metadata Management Technology: Status and Issues. *Journal of Object Technology*, 2005. **4**(2): p. 41-47.
6. Choudhury, S., et al. Loculus: A Metadata Wrapper for Digital Motion Pictures. in *Internet and Multimedia Systems and Applications (IMSA 2007)*. 2007. Honolulu, Hawaii, USA.
7. Yahoo. flickr. 2007 [cited 2007 March]; Available from: http://www.flickr.com/.
8. Smith, R. and B. Pham, A Robust Object Category Detection System using Deformable Shapes. *Machine Vision and Applications Journal 2007* (Accepted and awaiting publication.).
9. W3C. The Extensible Stylesheet Language Family (XSL). 2007 [cited 2007 March]; Available from: http://www.w3.org/Style/XSL/.